apple mechanic

by Bert Kersey

[illegible]

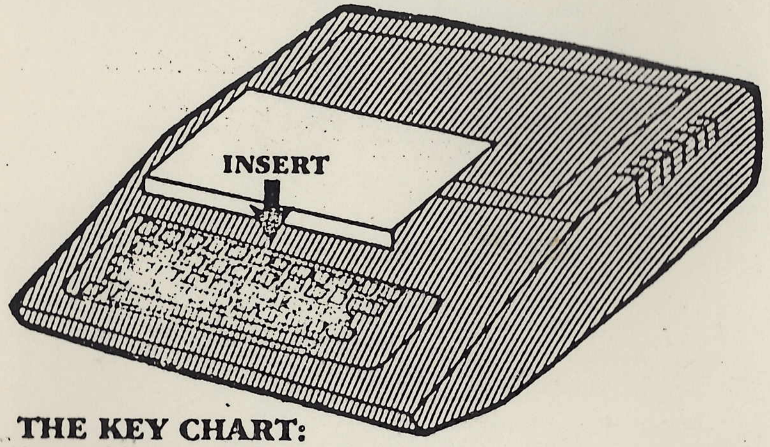
[illegible] or this disk, and as always, we hope you will find them both useful *and* entertaining. In using this disk, you will hopefully learn some new Apple programming techniques that will help you write better programs. With this in mind, all *Apple Mechanic* programs have remained listable and inspectable by normal methods. Let's get on with the instructions—

BACK UP YOUR DISK:

Unlike most commercially-made disks, *Apple Mechanic* is not copy protected (just copy*righted*) and supports normal Apple DOS. Either make a back-up with one of the copy programs on your *System Master* disk or use the *FID* program from the same disk. And store the original in a safe place, away from prying magnetic fields.

BOOTING:

Booting will not be necessary if you have recently booted a normal-DOS disk and haven't been doing any exotic fooling around. If a program on *Apple Mechanic* gives you a problem, try typing "FP" (return) and re-running the program, or go ahead and boot. For you newcomers out there, *boot* means turn your Apple off, insert your disk in Drive 1, close the door and turn the Apple on. Or leave the power on and type (usually) "PR#6" (return). When your drive's red light goes off, you and your Apple are ready to go to work.



THE KEY CHART:

A two-sided Key Chart has been included with your disk. FOLD IT LENGTHWISE with the appropriate side facing up, and insert it behind the top row of keys on your Apple. The Key Chart provides convenient labels for the "key" keys in several of the *Apple Mechanic* programs and puts the instructions in front of you where they will do the most good. Functions that are not specified on the Key Chart will appear on your monitor at the appropriate times. Fold and store your Key Chart with your disk. (Note: Leave your disk *un*folded; it'll work much 'er that way.)

SHAPE EDITOR (page 21)
Allows you to create 12-shape shape tables for use in your programs.

FONT EDITOR (page 31)
Allows you to edit and create hi-res character sets or "shape-fonts" to be used with your programs, *Xtyper* or *Hi-Writer*.

FONT SPLITTER (page 42)
Allows you to reduce the number of characters in any shape-font.

SHAPE ANALYZER (page 43)
Lets you analyze shape tables and test each shape in various combinations of ROT, SCALE and HCOLOR.

XTYPER (page 34)
Lets you type shape-font characters directly onto a hi-res image or blank screen. The finished image may then be saved to disk.

HI-WRITER (page 38)
Allows program-control of shape-fonts. You add your own Applesoft commands to make "live" graphic presentations.

HI-WRITER DEMO (page 38)
List this to see how to make your own graphic presentations.

SHAPE TABLE DEMO (page 27)
A demonstration of shape table commands and loading procedures.

GREETINGS
Our "talking heads" say hello, using most of the graphics and sound features on *Apple Mechanic* (and then some).

PAGE COPY (page 42)
Three EXEC programs that move images between pages 1, 2 & 3.

BYTE ZAP (page 49)
A handy disk detective that lets you rewrite any byte on a disk.

SONG SUBROUTINES (instructions on disk)
A few little tunes and noises to liven up your programs.

TEXT TRICKS (instructions on disk)
Some text screen gimmicks; we couldn't resist.

TWO-LINERS
Spectacular winning entries from Uncle Louie's famous contest.

SHAPES, @IMAGES AND]FONTS
These are binary picture files and shape tables. *Cursors* and *Demo Shapes* are shape tables used by the editing and demo programs. Hi-res image file names are preceded by an "@". File names that start with a "]" (shift-M) are shape-fonts. You may, of course, use your own system for naming your own files.

Creating Shape Tables

Shape Tables are excellent hi-res animation tools. Each "table" is simply a series of hi-res drawings stored in memory in the form of plotting and non-plotting vectors. Shapes in each shape table may be called to the screen by Applesoft's DRAW and XDRAW commands. While being rather abstract in concept, shape tables are relatively easy to use. Difficulty often comes in *writing* shape tables. One look at your Applesoft Manual will tell you why. *Apple Mechanic* has two programs that make writing shape tables a snap. To learn more about shape tables, their usage and structure, read *How to Use Shape Tables* starting on page 27.

TWO SHAPE EDITORS

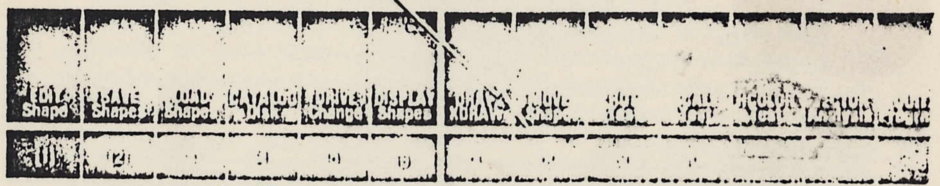
Two different shape-writing programs are on the *Apple Mechanic* disk. ***Shape Editor*** lets you create and store up to 12 shapes at a time, each one up to 48 x 63 plots in size. These shapes can be utilized by your own Applesoft programs. ***Font Editor*** lets you make hi-res characters or pictures that can be utilized by the *Xtyper* and *Hi-Writer* programs (both on this disk), as well as your own programs.

ERROR MESSAGES

Most error messages will relate to disk errors occuring when you are loading files. "FILE NOT FOUND" means the file you requested is not on the disk (or you spelled its name wrong) or you are using a disk that does not have the *Cursors* file on it. Other error messages will be referenced by number. See your Beagle Bros *Peeks & Pokes* Chart or Applesoft Manual for ONERR error message codes. If you suspect you have made an error, remove the ONERR statement from the beginning of the program you are working with. Then you can list the offensive line and get a clue regarding what's happening.

THE SHAPE TABLE KEY CHART

Use the Shape Editor side of the Key Chart that came with your *Apple Mechanic* disk. FOLD IT and insert it behind the top row of keys on your Apple. The 1-6 keys apply to the *Shape Editor* and *Font Editor* programs. The 3 through hyphen keys are for analyzing individual shapes with the *Shape Analyzer* program. Several Key Chart options are common to all three programs:



(2) SAVE EXISTING SHAPES

(Note: SAVE is not available from the *Shape Analyzer* program.)

After you have created one or more shapes with *Shape Editor* or *Font Editor*, your shapes are in your Apple's memory, but not on disk. To save your shapes onto disk, press the **2** key. You will be asked for a title for the shapes you are saving. Any legal file name is o.k., but it helps to add a descriptive word like "FONT" or "SHAPES" to each name to keep them straight. You can always rename files later.

You will recognize shape tables or shape-fonts in your catalog mainly by the title you have given them. There will also be a "B" for Binary to the left of the sector number. You cannot BRUN these files, only BLOAD them and then use them from your own Applesoft programs or *Xtyper* or *Hi-Writer*.

If you find yourself accidentally in the Save Mode, hit the RETURN key (with no file name) to escape, *or* enter one of the Key Chart keys (and return).

(3) LOAD NEW SHAPES

Select **3** to load a new shape table for editing or analyzing. The *Shape Analyzer* program can handle *any* Apple shape table, but *Font Editor* and *Shape Editor* are very specialized. Thus, this warning:

LOAD ONLY SHAPE TABLES MADE WITH THE EDITING PROGRAM YOU ARE USING. OTHERS WILL EITHER NOT LOAD OR NOT BE EDITABLE.

(4) CATALOG THE DISK

Back to the Key Chart— A **4** keypress lets you catalog the disk drive most recently specified; handy if you want to inspect your file names. You will be temporarily switched to all-text mode for the catalog. Hitting any key will then return you to hi-res plus text.

(5) DRIVE/SLOT CHANGE

Seleting **5** lets you change the drive that will next be accessed by the program. Hitting RETURN defaults to the slot or drive value printed on the screen. Do not select a drive or slot that is not connected or the program will hang.

The SHAPE EDITOR Program

This program lets you draw hi-res shapes by plotting them on the screen from the keyboard. It will then temporarily store the drawings in your Apple's memory. When finished, you can save the drawings to disk in the form of a 12 shape (maximum) shape table for use in your Applesoft programs.

ONLY SHAPE TABLES MADE WITH THIS PROGRAM CAN BE EDITED WITH THIS PROGRAM. OTHERS *MAY* LOAD, BUT WILL NOT BE COMPATIBLE OR EDITABLE.

To begin, insert the *Apple Mechanic* disk in your drive, and type "RUN SHAPE EDITOR". When the program title is on the screen, six Key Chart keys control the Editor:

1: EDIT a Shape
2: SAVE Existing Shapes*
3: LOAD New Shapes*
4: CATALOG Disk*
5: DRIVE/SLOT Change*
6: DISPLAY Shapes

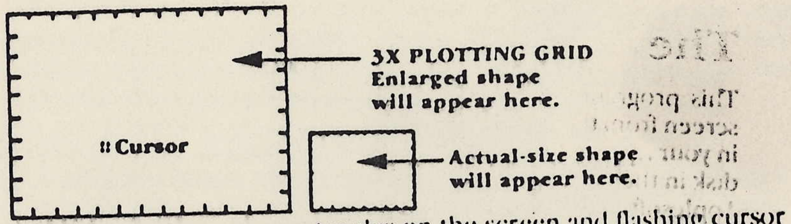
*Covered under *Key Chart*, page 23

You can usually "escape" to this part of the program by pressing the **ESC** key. When the program title is visible, you will see all 12 shapes currently in memory on the hi-res screen. If you are just starting, you will quite likely see the Editor's default shapes, 12 white dots. Each dot is actually a hi-res *shape* in a 12-shape shape table. Load a set of shapes, like *Demo Shapes*, as a test (press **3**, enter the name and wait while they load). You can *replace* each of the shapes you see with any drawing you want to create.

Due to the variety of shapes you might produce, the 12-shape menu display could get a bit messy. Unusually large shapes could overlap or "wrap around" and partially print on the opposite side of the screen. No problem, however; this display is for reference only. You can see each shape *alone* by selecting Key Chart Option **6** (more later) or by hitting **N** now to temporarily remove the text from the lower screen. Any key restores the text.

(1) EDIT A SHAPE

First, you need to choose which shape, 1-12, you are going to create. Actually you are going to *re-draw and replace* the shape you select. After pressing **1**, you will see a shape on the screen, perhaps a dot if you are just starting, both DRAWN and XDRAWN (these two often look the same), with its shape number below. To advance to the next or previous shape, use the Right or Left **Arrow** key. Shape #1 will follow #12 and vice versa. Another way to select a shape is to type the shape number you want, **1-9**. Shapes 10, 11 and 12 are selected for viewing/editing with the **Zero, Colon** and **Hyphen keys**. You can *animate* between any shapes you choose using these twelve top-row keys [illegible]xample, type 4-2-4-2-4-2-). With the shape or shape numbe[illegible]at you want to redraw on the screen, press RETURN.



Now you see two rectangles on the screen and flashing cursor made of dots. Notice that you can move the cursor left & right with the **Arrow** keys and up & down with the **A** and **Z** keys. You will be creating a shape in a magnified version (3 times actual size) by moving this cursor inside the larger rectangle. Simultaneously, you will see the shape plotted in actual size in the smaller rectangle. Move the cursor, if necessary, to its starting point (allowing room for your drawing) and hit RETURN.

Now you have a *solid* flashing cursor (with a dot in the middle representing your drawing's start point). A *solid* flashing cursor means that the next vector (dot) will **PLOT AND MOVE** (up, down, left or right). A *dotted* flashing cursor means the next vector will **MOVE ONLY AND *NOT* PLOT.** To switch between plot and move, press the SPACE BAR. Notice the word "PLOT" on the screen changes to "MOVE" and back. Draw your shape with the A, Z and Arrow keys. Your drawing will consist of a number of "vectors" (visible or invisible lines one-plot long). All vectors, plotted or not, *must move*.

Shape Editor limits shapes to the size of the plotting grid, 48 x 63 hi-res plots, and to 999 separate vectors. You will encounter some appropriate flashing and/or beeping if you approach these limits. The 999 is determined by the variable MAX near the beginning of the program listing. Change it if you want.

(X) ERASE

To erase parts of shapes as they are being drawn, use **X** as a "backspace" key. You cannot erase *pre-plot* dots with this feature.

(P) PRE-PLOT

It is best to plot each shape as efficiently as possible* using the fewest number of vectors. If you want, you can *pre-plot* your drawing on the screen without regard to efficiency and then *trace over* the sketch for your final drawing. *After* you have selected the shape to be drawn and positioned the cursor, and *before* you hit RETURN, select **P**. Now you can draw on the screen with a series of dots. The actual-size drawing in the smaller rectangle will appear as it normally would. To erase a dot, you need to plot on top of it (you can leave extra dots on the screen; they will have no effect on your final drawing). When finished pre-plotting, hit RETURN. The actual size sketch will disappear but the 3X version will remain. Now you can trace the dots to create your shape.

*One more note on efficiency: Don't give it too much thought. An "inefficiently" drawn shape will take up some extra memory all right and t[illegible]our Apple maybe a micro-second longer to draw. You can probably afford bot[illegible]

(I) IMPRINT

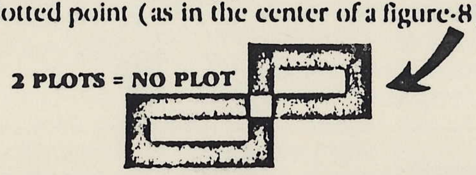
After you have selected the shape you want to edit, you may imprint any existing shape on the drawing screen by hitting **I**, selecting the shape to be imprinted (with the Arrow keys or 1-9, zero, colon or hyphen), and hitting RETURN. Imprinting is done if you want to create a shape similar to another (for animation, perhaps) by tracing over the imprint. The imprint is repositionable using the A, Z and Arrow keys. Repositioning is done to center the imprint in the rectangles or to align the imprint with your in-progress drawing.

To *keep* the imprinted image on the screen and start or continue plotting your shape, hit RETURN. To *erase* the imprint, hit ESC. The imprint will be erased from the small (actual size) rectangle, regardless of which option you select.

The imprint in the larger rectangle is XDRAWN with SCALE=3. You will notice that enlarged shapes seem to be a bit inaccurate. Ends of lines go too far or are bent at their ends, and dots appear as short lines. This is just the nature of scaled-up shapes. With a little practice, you'll get used to them and be able to use them as models for drawing. If the drawing you are creating is already in progress on the screen, you will see the imprint in *reverse* where it crosses your picture. In the actual-size drawing, the imprint and your drawing could actually cancel each other out temporarily.

PLOTTING TIPS

Lines that cross each other can be confusing at first. A plotting vector (solid cursor) will normally *plot*. But a vector that plots over an already-plotted point (as in the center of a figure-8) will *unplot* the point. You may ignore this problem if you are going to mainly DRAW your shapes from your programs, but if you are going to XDRAW or animate them, the flaw will show. To move the plotting cursor *without* changing the drawing under it, use the SPACE BAR to select the dotted (non-plotting) cursor. And remember, you can always correct mistakes by backing up with the **X** key.



Positioning the cursor before plotting is usually necessary only on *large shapes* that might approach the limits of the drawing rectangle. If you are planning a shape that is going to go off to the right, for example, you should position the cursor near the *left* edge of the rectangle *before plotting*, giving you room to draw. The cursor starting position will remain the same until you change it.

The last point of a shape is where the *next* shape will start if your program uses a command like "XDRAW 1: XDRAW 2". In this example, Shape # 2 will be drawn on the screen starting where Shape # 1 stopped. If you will always be using plotting coordinates in your progr s in "XDRAW 1 AT 10,10: XDRAW 2 AT 10,99", you don't need oncerned where your shapes end.

You may use less than 12 shapes by simply ignoring the ones you don't want. The default dot shapes take up only two bytes apiece. **To create shape tables with more than 12 shapes, use the *Font Editor* program, which allows 95 shapes.**

(RETURN) FINISHING A SHAPE

Usually, you won't care where your shape *ends*. Just draw until it looks the way you want (especially in the actual-size version), and hit RETURN. When asked, "READY TO WRITE SHAPE INTO TABLE?", answer **Y**. If you had answered **N**, you would have been allowed to continue drawing. If you want to start over, answer **N** and then **ESC**.

Now your drawing will be "written" into the shape table. The more complex your shape, the more time this will take. The main title will reappear when writing is complete. Create as many shapes as you like, from 1 to 12. Just select Option **1**, and select the shape you want to redraw next.

(6) DISPLAY EACH SHAPE

Selecting Key Chart Option **6** when the main title is visible will let you view each shape individually, DRAWN and XDRAWN. Use the **Arrow** keys to view different shapes, and RETURN to see all twelve.

How to Use Shape Tables

This section covers the basics of using shape tables. The commands are simple and the results are amazing! List the *Shape Table Demo* to learn much much more. And above all, **experiment!** Get in there and make some mistakes (heck, I can't teach you *everything*).

LOADING YOUR SHAPE TABLE

First, set ROT & SCALE to their minimum values, 0 and 1. Then decide where in your Apple's memory to load your shape table. For now, let's pick location 25,000; it's a nice round number and out of the way of other goings-on. Set a variable in your program (for example, SH) equal to 25000 and load your shape table like so—

```
10  ROT= 0: SCALE= 1
20 SH = 25000:D$ =  CHR$ (4)
30  PRINT D$;"BLOAD SHAPES,A";SH
```

A shape table is a Binary file, so it must be BLOADed (but never BRUN). To play it safe, always tell the shape table where to load by specifying the location as part of the BLOAD command. In the example above, you could have used "**,A25000**" for the same results. While programming, you won't want to wait while your shapes load every time you run the program, so (after you have run it once) put a temporary "REM" in front of your BLOAD statement.

On page 45, there is a discussion regarding *More Room for Hi-Res Programs*. If you load your Applesoft program, as suggested there, at location 24576 ($6000), you will need to load your shape table elsewhere or one will erase the other. Location 2048 ($800), the normal *program* location, would be appropriate and allow you the most space. If you like round numbers, set al to 3000.

SETTING THE SHAPE TABLE POINTER

Now you need to to tell your Apple where your shape table is (location 25,000) in memory. In your program, do two pokes—

```
40  POKE 232,SH -  INT (SH / 256) * 256
50  POKE 233, INT (SH / 256)
```

Assuming you have set SH equal to 25000, these two pokes put that number into the *Shape Table Pointer* at memory locations 232 & 233. If you run the above program now, that pointer will be set. If you *forget* to do the two pokes in your *next* shape table program, the program *might* work fine, because the pointer *stays set*, even surviving the often fatal "NEW" and "FP" commands. But tomorrow your Apple won't know where your shape table is stored. When you turn off your Apple or re-boot, the Shape Table pointer will "point" to the wrong place. The DRAW command will probably produce random scribblings, much as if you hadn't even loaded your shape table. So, *always* include the two pokes at the beginning of your shape table programs.

USING MORE THAN ONE SHAPE TABLE

To alternate between shape tables, simply re-poke a new location into the shape table pointer at 232-233 each time you change.

HCOLOR

To DRAW a shape, you need to specify a color by setting HCOLOR equal to a number, 0-7. If you don't, your shape might be drawn in HCOLOR 0 (black) and you won't see it. Here are Apple's hi-res colors.—

HCOLOR=0 or 4: Black
HCOLOR=1: Green
HCOLOR=2: Violet
HCOLOR=3 or 7: White
HCOLOR=5: Orange
HCOLOR=6: Blue

DRAW AND XDRAW

DRAW means just that, *draw* a shape from a shape table. Use the command like this—

```
99  DRAW 3 AT 100,90: REM (AT X,Y)
```

The first number after DRAW is the number, 1-255, of the shape to be drawn. The next two numbers are the horizontal and vertical hi-res coordinates of the starting point of the shape. If you get an ?ILLEGAL QUANTITY error, you have either attempted to draw outside of the screen's limits *or* you have specified a shape number larger than the number of shapes available.

The **XDRAW** command works like DRAW, but ignores HCOLOR and plots each dot of your shape in the *opposite* color of the dot that is being plotted over. XDRAW is handy for animation because you can [illegible] a shape travel over and be seen against multi-colored backgro[illegible].

BLACK (0/4) is opposite WHITE (3/7)
GREEN (1) is opposite VIOLET (2)
ORANGE (5) is opposite BLUE (6)

HI-RES COORDINATES

To tell a shape where to appear, specify the coordinates (horizontal, vertical) of the shape's starting point. Don't specify a start location outside of the screen limits, or your program will crash. Here are the hi-res limits for Apple hi-res—

Horizontal: 0 to 279
Vertical: 0 to 191
Vertical with 4 text lines: 0 to 159

DRAWING WITHOUT COORDINATES: After your first shape is drawn, you don't need to specify coordinates for the next DRAW or XDRAW, but you probably will want to. Each successive shape drawn without coordinates (as in "XDRAW 3: XDRAW 1: XDRAW 9") will begin *at the point where the previous shape stopped*. Eventually, using consecutive commands like this, shapes will usually "wrap around" and begin plotting on the opposite side of the screen.

SCALE

You can enlarge a shape by setting SCALE equal to a number, 2-255 or zero. A SCALE of zero is equivalent to a SCALE of 256. Scaled-up shapes are of limited use because the plotting vectors are *lengthened only* and spread apart as they are enlarged. They often end strangely-bent at right angles, depending on how the shape was created. In short, scaled-up shapes usually just don't look right.

ROT

You can rotate a shape by setting ROT before you draw—

ROT=0: Normal
ROT=16: Rotated 90° Clockwise
ROT=32: Rotated 180°(upside down)
ROT=48: Rotated 270°Clockwise

ROT values between the above only apply when SCALE is set larger than minimum (1). ROT values greater than 64 simply repeat the cycle until ROT=255.

MOVING A SHAPE

There is more than one way to move a shape. Basically you have to *draw* the shape, *erase* it, and *redraw* it in a new position. This XDRAW example moves a shape across the screen—

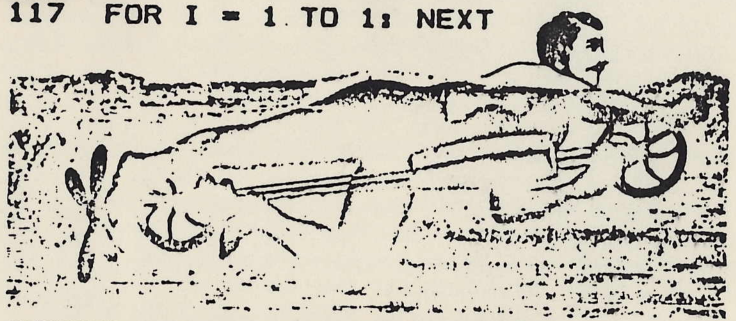
```
100  FOR X = 0 TO 279
110  XDRAW 5 AT X,100: REM DRAW SHAPE
115  XDRAW 5 AT X,100: REM ERASE SHAPE
120  NEXT X
```

With DRAW, you erase the shape by re-DRAWing it in the background color—

```
100  FOR X = 0 TO 279
110  HCOLOR= 3: DRAW 5 AT X,100: REM DRAW
115  HCOLOR= 0: DRAW 5 AT X,100: REM ERASE
120  NEXT X
```

Experiment with adding different delay loops between drawing and erasing. For example add these lines—

```
113  FOR I = 1 TO 50: NEXT
117  FOR I = 1 TO 1: NEXT
```



OTHER HI-RES MANIPULATIONS

HGR— Clears Page 1 to black & reveals it for drawing.
HGR2— Clears Page 2 to black & reveals it for drawing.
HCOLOR=X: HPLOT 0,0:
CALL 62454— Clears hi-res in HCOLOR X

The following "switches" do not clear the screen, but *reveal* whatever is currently on the page 1 or 2 text, hi-res or lo-res screens:

POKE 49232,0— View graphics screen* (hi- or lo-res)
POKE 49233,0— View text screen
POKE 49234,0— View full graphics screen (hi- or lo-res)
POKE 49235,0— View graphics plus 4 text lines* (VTAB 21-24)
POKE 49236,0— View Page 1* (hi-res, lo-res or text)
POKE 49237,0— View Page 2
POKE 49238,0— View lo-res
POKE 49239,0— View hi-res*

*Automatically set by HGR

POKE 230,32— Allows drawing on page 1
POKE 230,64— Allows drawing on page 2
POKE 230,96— Allows drawing on page 3 (not directly viewable)

The three pokes above disregard which page is currently being viewed. Poking other numbers into location 230 produce unpredictable, often disastrous results. Try POKE 230,48 and do some hplotting, for example. Be prepared to lose your program and re-boot if you try other numbers.

PRINT PEEK(228)— Prints HCOLOR code (see *Peeks & Pokes*)
PRINT PEEK(231)— Prints current SCALE value
PRINT PEEK(249)— Prints current ROT value
CALL 62923— Stores last plotted coordinates at locations 224-226
PEEK(6)— VERTICAL position during last CALL 62923
PEEK PEEK(225)*256— Horizontal during CALL 62923

The FONT EDITOR Program

This program is similar in concept to *Shape Editor*, but specialized to create shape table character sets or *fonts* of type. Each "shape-font" consists of 95 hi-res characters that can be printed onto the screen by adding Applesoft commands to the *HI-WRITER* program. Or you can type the characters directly onto a hi-res picture or blank screen by running the *XTYPER* program. Shape-fonts may be redesigned or used as-is without editing. Any character may be redrawn as you like (as pictures, symbols or whatever), as long as it fits within the maximum dimensions specified by the Editor.

ONLY SHAPE-FONTS MADE WITH THIS PROGRAM CAN BE EDITED WITH THIS PROGRAM. OTHERS *MAY* LOAD BUT WILL NOT BE COMPATIBLE OR EDITABLE.

SHAPE FONTS vs. OTHER FONTS:

In your encounters with other disks, you will undoubtedly come across other hi-res type fonts. *Apple Mechanic's* shape table type is unique, however, and incompatible with most other Apple type. That is, you can't use the same programs to display or edit shape-fonts as you do to display or edit other kinds of hi-res type.

ADVANTAGES: Shape-fonts have the distinct advantage of being *proportionally spaced*. An "I", for example, doesn't need to occupy the same screen space as a "W". Word-spacing and the distance between lines of type is adjustable. And type doesn't have to be any certain height; it can range between 3 and 16 dots high, exactly as you designed it, and be positioned *anywhere* on the hi-res screen without regard to vtabs or htabs. Shape-font type can be *XDRAWN* too, so that each character is automatically printed in the opposite of the background color. And shape-font characters are fast— *Font Editor* creates characters that will be drawn from upper-left to lower-right, ending adjacent to the left edge of the next character to be printed. Therefore coordinates do not need to be specified when displaying each character-shape.

DISADVANTAGES: Shape fonts take up more memory and disk space than other hi-res fonts you may have seen, usually around 18 sectors for large-height fonts and 7 for small-height fonts (full 95-character fonts). You will find fonts on Apple Computer's *DOS Tool Kit*, for example, that only occupy 5 sectors. The difference is in the flexibility and attractiveness of the typefaces and resulting screen presentations.

LARGE FONTS vs. SMALL FONTS

All shape-fonts are classified as Large or Small. Small fonts correspond approximately in height to Apple's normal text characters. All font names on the *Apple Mechanic* disk are preceded by a "]" (shift-M). All are large-height fonts, except those with the word "small" in their names. Another way to recognize font size is by catal sector size— Full 96-character large fonts' tend to be 16-20 se small fonts 6-10 sectors. Large-font characters may be up to 14 x 16 hi-res plots in size. Small characters are limited to 7 x 8. The difference is the alignment of characters when printed with *Xtyper* and *Hi-Writer*. You may, of course, create small characters in a large font, but not vice versa. If you know how shape tables are constructed, you might like to know that the second byte is set to 1 for small fonts and 2 for large fonts.

RUNNING THE PROGRAM

To begin, insert the same Key Chart you used for *Shape Editor*. Boot the disk, if necessary, and "RUN FONT EDITOR". When the program title is on the screen, six keys control the program:

1: EDIT a Character
2: SAVE Existing Shapes*
3: LOAD New Shapes*
4: CATALOG Disk*
5: DRIVE/SLOT Change*
6: DISPLAY (Type) Shapes

*Covered under *Key Chart*, page 23

You can usually escape to this part of the program by pressing the **ESC** key. See the *Shape Table Key Chart* notes on page 23 for details on options 2-5.

When you run the program, you will see the plotting grid and the 95 characters currently in memory. If you are just starting, you might see the Editor's default "characters", 95 white dots. Each dot is actually a hi-res *shape* in a 100-shape shape table. If you wanted a font with just a few characters (only numbers, for example), you could start with this dot "font" and add only your special characters, or you can use the *Font Splitter* program (page 42). More likely, you will want to load one of the complete fonts on the disk (using Key Chart Option 3) and change a few characters.

THE PLOTTING GRID

After you have loaded a font with Key Chart Option 3, the character set will be displayed. To the left you will see a rectangular plotting grid for drawing characters at 10 times actual size. The grid's size depends on whether a large (14 x 16) or small (7 x 8) font was most-recently loaded. The dotted lines on the grid have no function other than to serve as visual guides for lower case, descenders and so on. Their use is up to you.

(1) EDITING CHARACTERS

After you have loaded a font (Key Chart Option 3), select the character to be edited by pressing the **1** key. A striped flashing cursor will appear on one of the characters. To move the cursor to the character you want to edit or replace, use the **Arrows** and **A & Z** keys. Below the character set, you will see the ASCII value of the key that will eventually type the character (for example, KEY "A", ASCII 65). The SPACE character, ASCII 32, is the only one not accessible from the Editor. Hit RETURN when the cursor is on the character you want t

Now you have three options:

(I) GRID IMPRINT: Prints the selected character as a 10x blow-up on the grid; good for making minor changes since most of the character will already be drawn for you.

(E) ERASE/REDRAW: Temporarily erases the character from the screen so you can completely redraw it.

(S) SHADOW IMPRINT: Prints a "shadow" of a character on the 10x grid as a reference. The shadow is for reference or tracing purposes only and does not affect the final shape.

Select **I, E** or **S**, and stand by. You will now see a round dot cursor, either outlined or solid, flashing in the upper-left of the grid. The **A, Z** and **Arrow** keys will move the cursor within the confines of the grid. Pressing the **Space Bar** changes the cursor from *Solid* to *Outlined* and back, representing *Plot* and *No-Plot*. Play around to see how it works. Notice that the actual-size character in the font to the right changes as you plot on the grid. You won't have the problem of dots unplotting when you plot over them (as in *Shape Editor*), because you are not actually creating the shape yet, only sketching it. The actual shape vectors will be created by the Editor, so there is no need to plan an "efficient" plotting path. Continue plotting and unplotting until the character looks the way you want it.

The shape you have plotted is only temporary; it isn't in the Apple's memory yet, only on the screen. To place it in memory (into a shape table), hit RETURN, and sit back. A cursor will scan the grid and "read" the shape. Now you are asked to "SELECT CHARACTER TO REPLACE". The character you just drew can now replace *anyone* of the 95 characters on the screen. Usually you will want to replace the same character you first selected, so just hit RETURN to insert the new shape data into memory.

If you start with a dot (blank) font, and *if* you draw a character other than the first character (ASCII 33), and *if* you quit the program and re-run it, you will see your shape *repeated* from the first shape position up to the point where you inserted your new shape. Well, what you see is what you get. That is one of the quirks of working with the blank dot font, since the "font" actually consists of just *one dot* referenced by all 95 characters' pointers. To play it safe, when starting with the dot font, edit the *lowest* numbered character available first.

(6) DISPLAY (TYPE) SHAPES

Font Editor features a one-line character typer for testing the appearance of your characters adjacent to other characters. Select 6 from the main menu and type as you normally would. The **ESC** key toggles between upper & lower case. **RETURN** or typing near the right margin will return you to the main menu. Because of program space limitations, *the backspace key does not function* in this mode. There is no way to move the cursor without typing, or to save an image either; this mode is for test purposes only. To type and save hi-res images, use the *Xtyper* or *Hi-Writer* programs.

The XTYPER Program

This program may be used to type onto the Page One hi-res screen using up to three shape-fonts at a time. You may want, for instance, to label a diagram or graph that you have plotted with another program. After you have added type to your picture (or to a blank screen), you can save the entire image to disk. To start, type "RUN XTYPER" and be sure your disk has at least one large or small shape-font on it including every font you intend to use.

THE MAIN MENU

Running *Xtyper* or selecting **ctrl-R** while typing displays the main menu and lets you load fonts from the disk, clear the hi-res screen, load & save hi-res images, and quit the program. Exit the menu and begin typing by pressing RETURN.

(1, 2 & 3) LOADING FONTS

You may load up to three large or small shape-fonts into memory in any order you wish. Select **1**, **2** or **3** from the menu and enter the name of the font. The font will then be loaded into memory showing you its start location after the font name (for example, "BLOAD FONTNAME, A18880"). You don't need to be concerned with this number. You may replace any font any time you wish from the main menu. If you *re-load* Font 1 or Font 2, any higher-numbered font in memory will automatically be reloaded at a new location. Therefore all three of the fonts you are using need to be on the same disk. If you attempt to load Font 2 or Font 3 when there is a lower number unused, your font number will be changed to the lowest unused number.

(X) CLEAR HI-RES SCREEN

Always select **X** to clear the screen *if you have no image loaded on* hi-res Page One. Otherwise you could encounter a "snowy" background (an uncleared screen) on which to type. If this happens, type **ctrl-R, X, Y** and **RETURN** (just follow the screen prompts).

(L) LOAD HI-RES PICTURE

Select **L** to load a hi-res picture from your disk. This could be a chart, a graph, a picture of your dog, or any hi-res image to which you want to add hi-res type.

(S) SAVE EXISTING PICTURE

After you have typed on a cleared screen or on your picture, select **S** from the main menu to save the entire image on disk. If you don't want a previous picture to be erased, don't use the same name when re-saving it. You will recognize a picture file in your catalog with its name preceded by a "B" for Binary and (usually) an "034" for 34 sectors. It helps if you name picture files with the word "picture" or similar identification in the title. We have used an "@" before *Apple Mechanic* picture file names.

Warning: Be sure your shape-font files are *locked.* If you accidentally enter a *font* name when saving a hi-res image, the font will be erased from the disk if the font is not locked! You may lock and unlock files by first selecting **C** to catalog.

Note: To see any hi-res picture on Page One when you are not running a program, type **HGR** (return) and **BLOAD PICTURE, A$2000** (return), where "PICTURE" is the name of the image you want to see.

(C) CATALOG DISK (and Lock/Unlock)

Select **C** to catalog from the main menu. You will need to do this to check the names of fonts or hi-res pictures. You will be given the option of locking and unlocking files after the catalog.

(Q) QUIT PROGRAM

Selecting **Q** from the main menu lets you quit *Xtyper.* When you quit, you will see the message "GOTO 2000 TO CONTINUE". If you were to type "RUN" instead of "GOTO 2000", you would need to reload your fonts into memory.

(RETURN) BEGIN TYPING

Hit RETURN to begin or continue typing on the hi-res screen. You must have loaded at least one shape font to type.

TYPING WITH XTYPER

After you have loaded at least one font from the main menu and hit RETURN, you will see a flashing rectangular cursor on the screen that corresponds to the approximate height of the current font and the current case, upper or lower. Now you can type on the screen or use one of the control commands below. You will get best results if you type in white (any size) or in color (large fonts only) on the BLACK background areas of the screen (see *Color Change* below).

The following commands appear on your Key Chart:

(RETURN) CARRIAGE RETURN

Hitting **RETURN** while typing puts the cursor at the left margin on the next line down. To change the vertical distance the cursor will move, see *Leading Notes* below. If you continue typing to the right margin, your words will "wrap around" without dropping to the next line. With *Xtyper* you must hit RETURN just as you would on a typewriter.

(ESC) UPPER & LOWER CASE

The **ESC** key will toggle between upper & lower case. The cursor height will change accordingly. Since shape-fonts can vary in character height, the height of the cursor might not exactly match.

(ctrl-P, ctrl-T & ctrl-Q) NON-KEY CHARACTERS

Three Apple characters have no corresponding keys. With *Xtyper,* **ctrl-P** types a left-square bracket* (the *right*-square bracket* is typed with shift-M as usual). **Ctrl-T** types the underscore*, and **ctrl-Q** types the backslash.*

*Or the character programmed for that key.

MOVING THE CURSOR:

Eight commands move the cursor on the screen without typing or erasing. You must hold the **CTRL** key down while you move the cursor or a letter will be typed. If you have a spare finger, you may also use the **REPT** key for long distance moves. The Left & Right **Arrows** (ctrl optional) move the cursor left and right approximately *one character*. Move your fingers three keys to the left and use **ctrl-K & ctrl-L** to move the cursor left and right *one plot*. **Ctrl-A & ctrl-Z** move the cursor up and down approximately *one type line*. And one key to the right, use **ctrl-S & ctrl-X** to move the cursor up and down *one plot*. Due to the varying sizes of shape-font characters, the cursor height will not always exactly match the height of the characters being typed. The *top of the caps cursor*, however, will usually align with the *top of the letter* about to be typed or erased.

ERASING:

Proportional type has many advantages. One *disadvantage* is the difficulty of making corrections. *Xtyper* gives you several ways to erase (and typing a *space* over a character is *not* one of them). First, *the top of the cursor must be aligned vertically* with the top of a capital letter on the line of type on which you wish to make an erasure. Got it? Now, here are your options for erasing:

(ctrl-B) BACKSPACE/ERASE: Place the left of your cursor at the right of the character you want erased. Now use **ctrl-B** as a backspace/erase key. A vertical black* line will be drawn at the left edge of the cursor, and the cursor will move one plot to the left. Use the REPT key if you want (press CTRL, B *and* REPT), and continue ctrl-B-ing until you are through erasing.

(ctrl-E) CHARACTER ERASE: This method works only with white type on a black background— Align your cursor vertically as described above, and position its left edge so it *touches* any part of the character to be erased. Now type **ctrl-E** and your Apple will erase the character (in black). *Stencil Font* characters will often only be partially erased with ctrl-E due to their split design.

(ctrl-W) WIPE OUT: Typing **ctrl-W** will erase an entire line of type in black* from the top of the caps cursor down.

OVERTYPING: Another way to erase a character is to place the cursor over it, aligning the top and left edges, change the typing color (ctrl-O) to the color of the background, and *type over* the characters to be erased using the same characters and the same font.

*See *D* [illegible] *rase Color* next page.

(ctrl-F) FONT CHANGE:

Selecting **ctrl-F** while typing will allow you to select a new typing font. You may not, of course, select a font number that has not been loaded into memory.

(ctrl-O) COLOR CHANGE:

Selecting **ctrl-O** while typing will allow you to select a new typing color. Due to Apple's hardware configuration, typing in colors other than white or black might present some surprises— Characters with thin verticals could look distorted, especially on a non-color monitor. Also, small characters are unreadable in colors other than black or white. And colored type or lines on a color background will tend to "vibrate" and produce "stair-stepped" edges. Apple's hi-res hardware is really designed to support colors on a BLACK BACKGROUND. *Xtyper* will not let you choose alternate color #7 (white) or #4 (black).

(ctrl-G) GRID:

Selecting **ctrl-G** while typing will display a 4 x 4 grid on your drawing. Hitting ctrl-G again will erase the grid. The purpose of the grid is to display the edges of the hi-res screen and to give some visual reference for aligning and centering type.

Warning: Don't type over the grid if you are going to erase the grid. If you do, you will draw (XDRAW) lines through characters that have come into contact with it.

(ctrl-R) RETURN TO MAIN MENU:

Selecting **ctrl-R** while typing will display the main menu, so you can save or load a picture, load a new font, clear the screen, catalog the disk or quit the program.

LEADING NOTES:

Leading (pronounced *"ledding"*) is the vertical distance between lines of type. *Xtyper* normally puts two lines of leading between characters, large and small. If you type a lower case "g" at the left margin, hit RETURN, and type an upper case "E", you will see a vertical gap of two plots between the characters. To change this default value of 2 to any other value, change the first non-REM program line in *Xtyper*. The variables LD(1), LD(2) and LD(3) represent the leading for the three fonts in memory. If you want *for* example, *four* plots of leading when you hit RETURN with Font 2, make LD(2)=4. Note: Apple's normal text uses a corresponding leading of *zero*, putting one vertical space between characters, however, because the standard text font has no descenders (parts of letters like *g, j, p, q* and *y* that go below the base line).

DEFAULT ERASE COLOR:

Just in case you are doing a lot of typing on non-black backgrounds, you may change the second non-REM program line in *Xtyper*. The variable BG is the erase color used by ctrl-B and ctr

Apple //e Notes

Due to the Apple //e's new top-row key-arrangement, there are some changes...

ALIGN YOUR KEYBOARD CHART LEFT/RIGHT WITH THE 1 THROUGH 0 KEYS, AND THE CHART'S COMMANDS WILL BE CORRECT.

```
-------------------------------------------
 1  2  3  4  5  6  7  8  9  0  :  -  RES
 1  2  3  4  5  6  7  8  9  0  -  =  DEL
 ^  ^  ^  ^  ^  ^  ^  ^  ^  ^
```

The Byte Zap WRITE-TO-DISK key is the EQUAL-key (keychart reads o.k.).

The Shape Analyzer Program now uses the HYPHEN, EQUAL and DELETE keys instead of the Colon, Hyphen and Reset keys (keychart reads o.k.).

AND...

If a program is responding oddly, check to see that your CAPS-LOCK key is DOWN.

GENERAL NOTES:

- SINCE THE APPLE MECHANIC DISK IS 0 % FULL, YOU MAY WANT TO USE FID (ON YOUR SYSTEM MASTER DISK) TO SPLIT IT INTO 2 OR MORE DISKS, THUS PREVENTING DISK FULL ERRORS WHEN SAVING FILES.
- THE FLUSH-LEFT TITLES AND BLANK FILE NAMES IN THE CATALOG WERE MADE WITH BYTE ZAP AND WILL TRANSFER WITH FID.
- IF YOU SUSPECT AN ERROR IN A PROGRAM, REMOVE THE ONERR GOTO STATEMENT FROM THE LISTING AND RE-RUN THE PROGRAM FOR A REPORT ON ERROR TYPE AND ERROR LINE NUMBER.
- TO MAKE A HARD COPY PRINTOUT OF A HI-RES IMAGE, YOU NEED TWO THINGS-- (SEE YOUR APPLE DEALER.)
  1. A PRINTER WITH HI-RES CAPABILITIES.
  2. HI-RES DUMP SOFTWARE THAT IS COMPATIBLE WITH YOUR PRINTER.

BYTE ZAP NOTES:

* THE GREETING (HELLO) PROGRAM NAME ON APPLE MECHANIC IS APPLE MECHANIC. TO CHANGE IT, SEE PAGE 34.

* THE DISK VOLUME CATALOG HEADING HAS BEEN CHANGED TO BEAGLE BROS. TO CHANGE IT, SEE PAGE 58.

* TO MAKE THE VOLUME NUMBER PRINT OR NOT PRINT WHEN YOU CATALOG, YOU CAN MAKE A CHANGE ON TRACK 1, SECTOR 12 ($0C):

BYTE NUMBER:	PRINT VALUE:	NO-PRINT VALUE:
192 ($C0),	32 ($20)	234 ($EA)
193 ($C1),	66 ($42)	234 ($EA)
194 ($C2),	174 ($AE)	234 ($EA)

* REMEMBER, DOS CHANGES WILL ONLY BE IN EFFECT *AFTER BOOTING* THE CHANGED DISK.

NEW FEATURES:

* TO RE-READ THE CURRENT TRACK & SECTOR WITH BYTE ZAP, PRESS THE *SPACE BAR*.

* A THIRD PAGE COPY PROGRAM (SEE PAGE 42) HAS BEEN ADDED TO THE DISK.

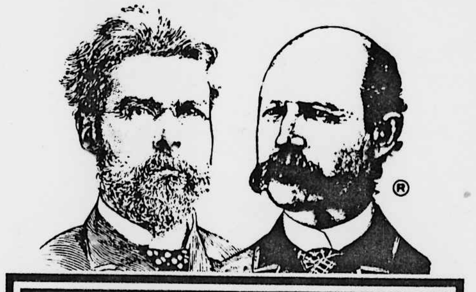
`EXEC PAGE COPY 1=3` MOVES PG.3 TO PG.1.

`EXEC PAGE COPY 1=3,R4` MOVES PG.1 TO 3.

SEE PAGE 30 (POKE 230,ETC.) IF YOU ARE UNFAMILIAR WITH HI-RES PAGE 3.

* THE XTYPER PROGRAM NOW SUPPORTS *TWO* DISK DRIVES.

REMEMBER TO FOLD YOUR KEY CHART LENGTHWISE AND INSERT IT BEHIND THE TOP ROW OF KEYS ON YOUR APPLE.



Beagle Bros
MICRO SOFTWARE

Apple Mechanic

COPYRIGHT © 1982, BERT KERSEY, BEAGLE BROS, INC.
3990 OLD TOWN AVENUE, SAN DIEGO, CALIFORNIA 92110

XTYPER
(Direct typing of Shape Fonts)

FUNCTIONS—
- Return: Carriage Return
- Esc: Upper/Lower Case
- Ctrl-F: Font Change
- Ctrl-G: Grid On/Off
- Ctrl-O: Color Change
- Ctrl-R: Return to Main Menu

ERASING—
- Ctrl-B: Backspace/Erase
- Ctrl-E: Character Erase (B&W)
- Ctrl-W: Wipe Out a Line

CURSOR MOVES—
- Arrows: Left/Right apx. 1 character
- Ctrl-K/L: Left/Right 1 hi-res plot
- Ctrl-A/Z: Up/Down apx. 1 line
- Ctrl-S/X: Up/Down 1 hi-res plot

HI-WRITER
(Program-controlled Shape-Fonts)

GOSUB 100: Execute
A$: Copy to be Printed
FT: Current Font (1-3)
FT3$: New Font 3 to Load
VT: Vertical Position (1-24)
HT: Horizontal Position (1-40)
IN: Inverse (1-Yes, 0-No)*
CT: Auto-Center (1-Yes, 0-No)*
HC: Type HCOLOR (0-7)
CL: Clear Screen HCOLOR (0-7)
RT: ROT Value/16 (0-3)*

POKE TXT,0: Partial Text
POKE FULL,0: Full Graphics
TEXT: Full Text
POKE HIRES,0: Hi-Res

*Reset to zero after every GOSUB 100

—Shape Editor/Font Editor—

					(Font Editor: Typing Mode) Arrows: ±1 A/Z: ±10
EDIT Shape	SAVE Shapes	LOAD Shapes	CATALOG Disk	DRIVE Change	DISPLAY Shapes
—1—	—2—	—3—	—4—	—5—	—6—

—Shape Analyzer—

	Arrows/A/Z: ±9 K/L/S/X: ±1	Arrows: ±16 A/Z: ±1 Return: Rot=0	Arrows: ±1 A/Z: ±20 Return: Scale=1	Arrows: ±1 H: Shape Color B: Background Return: B&W		
DRAW XDRAW	MOVE Shape	ROT Test	SCALE Test	HCOLOR Test	VECTOR Analysis	QUIT Progra
—7—	—8—	—9—	—0—			